Network Operads

Joe Moeller
moeller@math.ucr.edu

—

(joint w/ J. C. Baez, J. D. Foley, and B. S. Pollard)

2017 AMS Sectional Meeting at UCR

Constructing Network Operads

In the previous talk, John Foley talked about algebras of network operads.

Constructing Network Operads

In the previous talk, John Foley talked about algebras of network operads. Now we will see how network operads are constructed.

Constructing Network Operads

In the previous talk, John Foley talked about algebras of network operads. Now we will see how network operads are constructed.

- Start with your favorite lax symmetric monoidal functor $F\colon \mathbf{S} \to \mathbf{Cat}$
- apply the symmetric monoidal Grothendieck construction to get the symmetric monoidal category $\int F$ with $\otimes_F$
- Let O_F be the endomorphism operad $\mathrm{op}(\int F)$

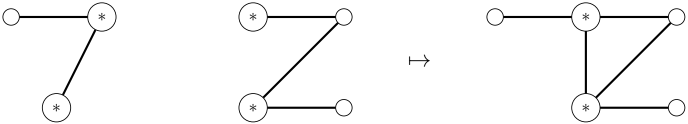
Theorem

The composite

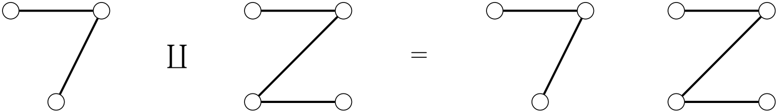
$$\mathbf{NetMod} \xrightarrow{\int} \mathbf{SSMC} \xrightarrow{\mathrm{op}(-)} \mathbf{Op}$$

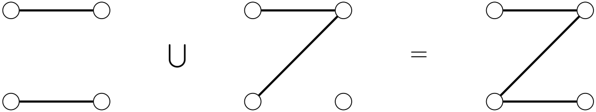
constructs a network operad for each network model.

Graphs can be combined to create bigger graphs by identifying some of the vertices

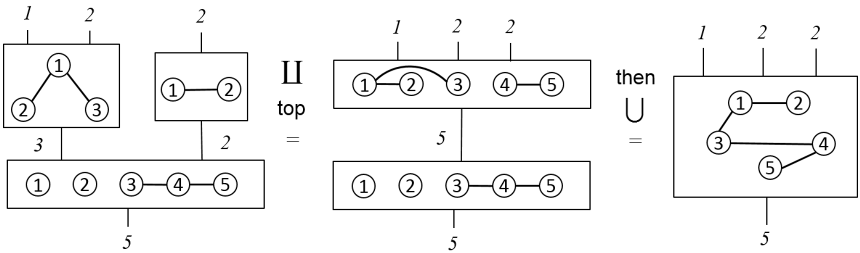


We choose to examine these as combinations of a few simpler operations





We want to construct an operad that captures these operations



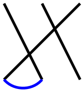
Endomorphism Operad

Given a symmetric monoidal category $(\mathbf{C}, \otimes)$, one can get a typed (aka coloured) operad $\mathrm{op}(\mathbf{C})$, called the **endomorphism operad**, with

- objects of $\mathbf{C}$ as types
- $\mathrm{op}(\mathbf{C})(c_1, \ldots, c_n; c) = \mathrm{Hom}_{\mathbf{C}}(c_1 \otimes \cdots \otimes c_n, c)$

Graph-Permutation Pairs

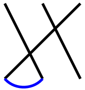
A morphism $(\sigma, g)\colon n \to n$ is a graph-permutation pair

(τ, h) 

Graph-Permutation Pairs

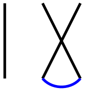
How should we compose two such pairs?

(τ, h)



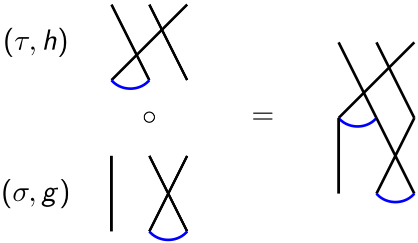
$\circ$

(σ, g)



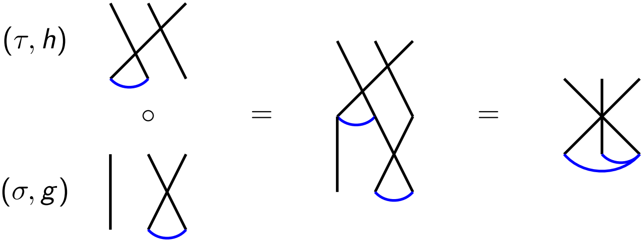
Graph-Permutation Pairs

How should we compose two such pairs?



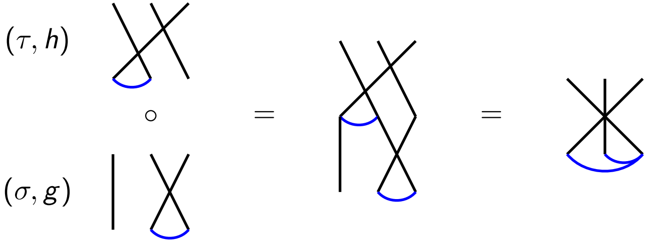
Graph-Permutation Pairs

How should we compose two such pairs?



Graph-Permutation Pairs

How should we compose two such pairs?

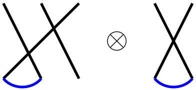


$$(\sigma, g) \circ (\tau, h) = (\sigma\tau, g \cup \sigma h)$$

Graph-Permutation Pairs

How should we tensor two such pairs?

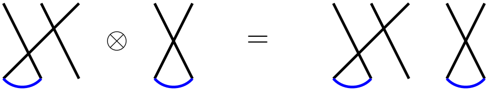
(σ, g) (τ, h)



Graph-Permutation Pairs

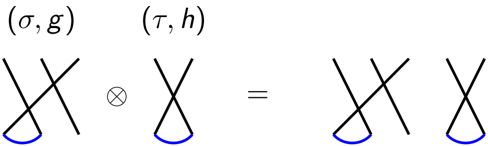
How should we tensor two such pairs?

$(\sigma, g) \qquad (\tau, h)$



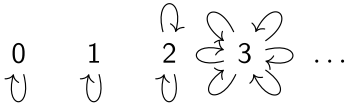
Graph-Permutation Pairs

How should we tensor two such pairs?



$$(\sigma, g) \otimes (\tau, h) = (\sigma + \tau, g \sqcup h)$$

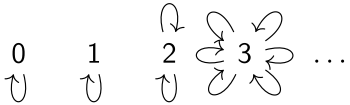
The Permutation Groupoid



Definition

Let $\mathbf{S}$ *denote the category with finite sets* $\mathbf{n} = \{1, \ldots, n\}$ *as objects, and bijections for morphisms.*

The Permutation Groupoid



Definition

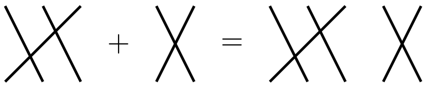
Let $\mathbf{S}$ *denote the category with finite sets* $\mathbf{n} = \{1, \ldots, n\}$ *as objects, and bijections for morphisms.*

Another way to see it is $\mathbf{S} = \coprod_{n \in \mathbb{N}} S_n$.

The Permutation Groupoid

S is a symmetric monoidal category with $+$ where

- $n + m$ is exactly what you think
- $\sigma + \tau$ looks like

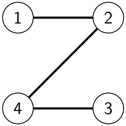


Simple Graphs

A simple graph with vertex set $\mathbf{n} = \{1, \ldots, n\}$ is a collection of subsets of $\mathbf{n}$, each of which have 2 elements.

Simple Graphs

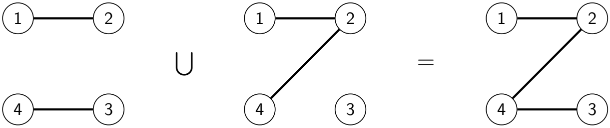
A simple graph with vertex set $\mathbf{n} = \{1, \ldots, n\}$ is a collection of subsets of $\mathbf{n}$, each of which have 2 elements. For example, this graph on $\mathbf{4}$



is the set $\{\{1,2\},\{2,4\},\{3,4\}\}$ in this setting.

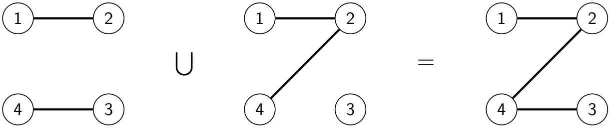
Graph Monoids

Defining graphs this way allows us to take unions of graphs



Graph Monoids

Defining graphs this way allows us to take unions of graphs



so the set of simple graphs on **n**, denoted $\mathrm{SG}(\mathbf{n})$, is a monoid with operation $\cup$.

S_n acts on the monoid $\mathrm{SG}(\mathbf{n})$ by permuting the vertices.

S_n acts on the monoid $\mathrm{SG}(\mathbf{n})$ by permuting the vertices. So we have a functor for each $n \in \mathbb{N}$

$$\mathrm{SG}_n \colon S_n \to \mathbf{Mon}$$

S_n acts on the monoid $\mathrm{SG}(\mathbf{n})$ by permuting the vertices. So we have a functor for each $n \in \mathbb{N}$

$$\mathrm{SG}_n \colon S_n \to \mathbf{Mon}$$

Putting them all together gives a functor

$$\mathrm{SG} \colon \mathbf{S} \to \mathbf{Mon}$$

Given two graphs, g in $\mathrm{SG}(\mathbf{n})$ and h in $\mathrm{SG}(m)$, the disjoint union $g \sqcup h$ is a graph in $\mathrm{SG}(n+m)$.

Given two graphs, g in $\mathrm{SG}(\mathbf{n})$ and h in $\mathrm{SG}(m)$, the disjoint union $g \sqcup h$ is a graph in $\mathrm{SG}(n+m)$. This gives a map

$$\sqcup \colon \mathrm{SG}(\mathbf{n}) \times \mathrm{SG}(m) \to \mathrm{SG}(n+m)$$

Given two graphs, g in $\mathrm{SG}(\mathbf{n})$ and h in $\mathrm{SG}(m)$, the disjoint union $g \sqcup h$ is a graph in $\mathrm{SG}(n+m)$. This gives a map

$$\sqcup \colon \mathrm{SG}(\mathbf{n}) \times \mathrm{SG}(m) \to \mathrm{SG}(n+m)$$

which makes SG a symmetric lax monoidal functor

$$(\mathrm{SG}, \sqcup) \colon (\mathbf{S}, +) \to (\mathbf{Cat}, \times)$$

The Grothendieck Construction

The Grothendieck construction takes a pseudofunctor

$$F \colon \mathbf{C} \to \mathbf{Cat}$$

The Grothendieck Construction

The Grothendieck construction takes a pseudofunctor

$$F\colon \mathbf{C} \to \mathbf{Cat}$$

and produces a category fibred over $\mathbf{C}$

$$\begin{array}{c} \int F \\ \downarrow \\ \mathbf{C} \end{array}$$

The Grothendieck Construction

In this talk, we will only be talking about functors

$$F \colon \mathbf{C} \to \mathbf{Cat}$$

The Grothendieck Construction

In this talk, we will only be talking about functors

$$F\colon \mathbf{C} \to \mathbf{Cat}$$

and the total category in the fibration

$$\int F$$

The Grothendieck Construction

Given a functor $F\colon \mathbf{C} \to \mathbf{Cat}$ the Grothendieck construction gives a category $\int F$ where

- objects are pairs (c, x) where
  - c is an object in $\mathbf{C}$
  - x is an object in Fc
- morphisms are $(f, g)\colon (c, x) \to (d, y)$ where
  - $f\colon c \to d$ in $\mathbf{C}$
  - $g\colon Ff(x) \to y$ in Fd
- composition is given by

$$(f, g) \circ (f', g') = (f \circ f', g \circ Ff(g'))$$

The Monoidal Grothendieck Construction

Let $(\mathbf{C}, \otimes)$ be a monoidal category, and $F\colon \mathbf{C} \to \mathbf{Cat}$ a lax monoidal functor, with $\Phi_{c,d}\colon Fc \times Fd \to F(c \otimes d)$.

The Monoidal Grothendieck Construction

Let $(\mathbf{C}, \otimes)$ be a monoidal category, and $F \colon \mathbf{C} \to \mathbf{Cat}$ a lax monoidal functor, with $\Phi_{c,d} \colon Fc \times Fd \to F(c \otimes d)$. Then we can define a monoidal structure on $\int F$ by

$$(c, x) \otimes_F (d, y) = (c \otimes d, \Phi_{c,d}(x, y))$$

and

$$(f, g) \otimes_F (f', g') = (f \otimes f', \Phi_{d,d'}(g, g'))$$

The Symmetric Monoidal Grothendieck Construction

Theorem

If $F : \mathbf{C} \to \mathbf{Cat}$ *is a lax monoidal functor, there is a natural way to define a monoidal structure on* $\int F$.

The Symmetric Monoidal Grothendieck Construction

Theorem

If $F\colon \mathbf{C} \to \mathbf{Cat}$ *is a lax monoidal functor, there is a natural way to define a monoidal structure on* $\int F$*.*

Theorem

If $F\colon \mathbf{C} \to \mathbf{Cat}$ *is a braided lax monoidal functor, there is a natural way to define a braided monoidal structure on* $\int F$*.*

The Symmetric Monoidal Grothendieck Construction

Theorem

If $F \colon \mathbf{C} \to \mathbf{Cat}$ is a lax monoidal functor, there is a natural way to define a monoidal structure on $\int F$.

Theorem

If $F \colon \mathbf{C} \to \mathbf{Cat}$ is a braided lax monoidal functor, there is a natural way to define a braided monoidal structure on $\int F$.

Theorem

If $F \colon \mathbf{C} \to \mathbf{Cat}$ is a symmetric lax monoidal functor, there is a natural way to define a symmetric monoidal structure on $\int F$. We call this the **symmetric monoidal Grothendieck construction**.

What does this mean for simple graphs?

We said before that we want to think about simple graphs as a symmetric lax monoidal functor

$$F \colon \mathbf{S} \to \mathbf{Cat}$$

$$\sqcup \colon F(n) \times F(m) \to F(n+m)$$

What does this mean for simple graphs?

We said before that we want to think about simple graphs as a symmetric lax monoidal functor

$$F \colon \mathbf{S} \to \mathbf{Cat}$$

$$\sqcup \colon F(n) \times F(m) \to F(n+m)$$

so we can apply the symmetric monoidal Grothendieck construction to F

What does this mean for simple graphs?

In this case, $\int F$ has

- objects are pairs $(n, *)$, with n an object in $\mathbf{S}$, and $*$ the only object in the monoid $F(n)$

What does this mean for simple graphs?

In this case, $\int F$ has

- objects are pairs $(n, *)$, with n an object in $\mathbf{S}$, and $*$ the only object in the monoid $F(n)$, so we just call it n

What does this mean for simple graphs?

In this case, $\int F$ has

- objects are pairs $(n, *)$, with n an object in $\mathbf{S}$, and $*$ the only object in the monoid $F(n)$, so we just call it n
- morphisms are (σ, g) where $\sigma \in S_n$, and $g \in F(n)$, ie a permutation and a graph

What does this mean for simple graphs?

In this case, $\int F$ has

- objects are pairs $(n, *)$, with n an object in $\mathbf{S}$, and $*$ the only object in the monoid $F(n)$, so we just call it n
- morphisms are (σ, g) where $\sigma \in S_n$, and $g \in F(n)$, ie a permutation and a graph
- composition is given by

$$(\sigma, g) \circ (\tau, h) = (\sigma\tau, g \cup \sigma h)$$

What does this mean for simple graphs?

In this case, $\int F$ has

- objects are pairs $(n, *)$, with n an object in $\mathbf{S}$, and $*$ the only object in the monoid $F(n)$, so we just call it n
- morphisms are (σ, g) where $\sigma \in S_n$, and $g \in F(n)$, ie a permutation and a graph
- composition is given by

$$(\sigma, g) \circ (\tau, h) = (\sigma\tau, g \cup \sigma h)$$

THIS IS THE COMPOSITION WE HAD BEFORE!

What does this mean for simple graphs?

The tensor in $\int F$ is

- $n \otimes_F m = n + m$

What does this mean for simple graphs?

The tensor in $\int F$ is

- $n \otimes_F m = n + m$
- $(\sigma, g) \otimes_F (\tau, h) = (\sigma + \tau, g \sqcup h)$

What does this mean for simple graphs?

The tensor in $\int F$ is

- $n \otimes_F m = n + m$
- $(\sigma, g) \otimes_F (\tau, h) = (\sigma + \tau, g \sqcup h)$

THIS IS THE TENSOR WE HAD BEFORE!

Summary of the construction

- Start with your favorite lax symmetric monoidal functor, $F\colon \mathbf{S} \to \mathbf{Cat}$, for example SG

Summary of the construction

- Start with your favorite lax symmetric monoidal functor, $F \colon \mathbf{S} \to \mathbf{Cat}$, for example SG
- apply the symmetric monoidal Grothendieck construction to get the symmetric monoidal category $\int F$ with $\otimes_F$

Summary of the construction

- Start with your favorite lax symmetric monoidal functor, $F\colon \mathbf{S} \to \mathbf{Cat}$, for example SG
- apply the symmetric monoidal Grothendieck construction to get the symmetric monoidal category $\int F$ with $\otimes_F$
- Let O_F be the endomorphism operad $\mathrm{op}(\int F)$

Now we can generalize this

Definition

A **Network Model** *is a lax symmetric monoidal functor* $F\colon \mathbf{S} \to \mathbf{Cat}$. *Let* **NetMod** *denote the category of network models.*

Theorem

The construction

$$\mathbf{NetMod} \xrightarrow{\mathrm{op}(\int -)} \mathbf{Op}$$

is functorial.

Examples of Network Models

- Multigraphs
- Directed Graphs
- Partitions
- Graphs with colored vertices
- Petri Nets

Examples of Network Models

- Multigraphs
- Directed Graphs
- Partitions
- Graphs with colored vertices
- Petri Nets
- Graphs with edges weighted by a monoid

Further reading and acknowledgments

- Check our e-print: Network Models, arXiv:1711.00037.
- You out for postings of technical reports [2, 3] for more examples.

This work was supported by the DARPA Complex Adaptive System Composition and Design Environment (CASCADE) project.

We thank Chris Boner, Tony Falcone, Marisa Hughes, Joel Kurucar, Tom Mifflin, John Paschkewitz, Thy Tran and Didier Vergamini for many helpful discussions.

- J. C. Baez, J. D. Foley, J. Moeller and B. S. Pollard, Network Models, Preprint, Available as arXiv:1711.00037.
- J. C. Baez, J. D. Foley, J. Moeller and B. S. Pollard, Operads for communication networks, 2016, 37 pages, Technical report, DARPA CASCADE project.
- J. C. Baez, J. D. Foley, J. Moeller and B. S. Pollard, Compositional tasking, 2017, 27 pages, Technical report, DARPA CASCADE project.
- N. Gambino and A. Joyal, On operads, bimodules and analytic functors, *Memoirs AMS* **249**, 2017. Available as arXiv:1405.7270
- F. W. Lawvere, Qualitative distinctions between some toposes of generalized graphs, *Contemp. Math.* **92** (1989) pp. 261-299.

D. Yau, *Colored Operads*, American Mathematical Society, Providence, Rhode Island, 2016.